RetroMagazine

RetroMagazine
Anno 1 - Numero 1

OTTOBRE 2017

SOMMARIO

IN EVIDENZA IN QUESTO NUMERO

Ridisegnamo i caratteri del C64

Un breve ma esaustivo tutorial per imparare l'ABC della programmazione BASIC, la mappatura della memoria video del C64 e per scoprire un software semi-sconosciuto ma utilissimo.

Articolo a pagina 2

IK+ (Amiga)

Un gioco di combattimento che e' entrato nella leggenda dei picchiaduro. Una pietra miliare che merita di essere giocata almeno una volta nella vita.

Articolo a pagina 5

Editoriale

di Francesco Fiorentini

Benvenuti nel 1986... anzi no! Benvenuti al primo numero di RetroMagazine, una rivista tutta italiana dedicata al mondo del retrocomputing scritta da appassionati per appassionati.

E' da un po' di tempo ormai che accarezzavo la romantica idea di una pubblicazione tutta incentrata sul mondo retro e finalmente ecco il primo numero!

Probabilmente anche molti di voi, come me, avranno visto in edicola delle riviste patinate e ben fatte dedicate ai computer ed ai giochi della nostra infanzia (parlo per quelli che, come me, hanno purtroppo gia' doppiato gli 'anta ☺), ma per un motivo o per un altro ne sono rimasti delusi. Quindi da qui l'idea di scrivermela da solo e condividerla con chi ha la mia stessa passione, sperando sotto sotto, di esortarlo a collaborare e rendere cosi' questa rivista piu' grande, migliore, bella e piu' ...nostra!

Molte delle riviste che ho trovato in edicola si concentravano principalmente sull'aspetto ludico del retrocomputing, tralasciando in parte o del tutto l'aspetto tecnico e la programmazione delle macchine a noi tanto care. Leggendo i vari blog, newsgroup, ed i gruppi facebook dedicati, ho scoperto di non essere il solo interessato anche all'aspetto tecnico e quindi vi preannuncio che il taglio dlela rivista andra' proprio in questa direzione, senza pero' tralasciare i giochi ed il divertimento!

Arrivato ormai al mezzo del cammin di nostra vita, mi son ritrovato in una selva oscura... ops, mi sono ritrovato a riscoprire il piacere di conoscere e capire come macchine favolose come il C64, lo Spectrum, l'Amiga... venivano programmate e... mi si e' aperto un mondo! Un mondo che vorrei condividere con voi.

Lungi da me il millantare competenze e l'arroganza di insegnare a chi ne sa' piu' di me; il mio umile intento e' solo quello di condividere con voi parte delle mie scoperte e perche' no, di scoprire qualcosa insieme.

Beh, credo di essermi dilungato anche troppo in questo primo editoriale, vi chiedo soltanto di apprezzare lo sforzo e per chi ne avesse voglia e tempo, di contattarmi per aiutarmi in questa avventura.

Come dice qualcuno piu' preparato di me, lunga vita al Commodore 64! Ed io aggiungo, anche all'Amiga, lo Spectrum, l'MSX...

C64List

da TXT a PRG e viceversa...

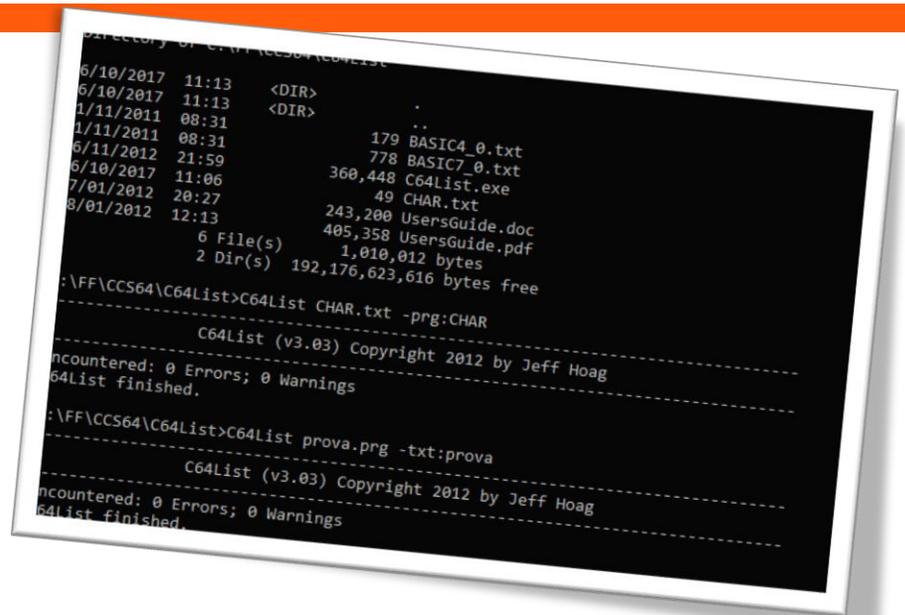
Il codice presentato nell'articolo sotto puo' essere scritto direttamente sul proprio C64 oppure, per chi non ne possiede uno, su un qualsiasi emulatore; ne esistono ormai svariati che emulano il C64 perfettamente.

VICE per esempio, ha una ottima funzionalita' di copia e incolla, ma per chi non puo' o non vuole usare VICE , preferendo altri emulatori come il CCS64, trovera' molto utile il programma che sto per presentarvi.

C64List e' un programma Windows (funzionante da riga di comando) che tra le svariate funzionalita', permette di convertire un file .TXT scritto con notepad in un file .PRG pienamente compatibile con gli emulatori ed il Commodore 64.

Questa e' la sintassi di C64List: **`C64list <input filename> [options]`**

Nel manuale allegato al programma ci sono riportate tutte le opzioni con la relativa spiegazione, ma nell'intento di questo articolo ci concentreremo solo su due di queste.

`-prg[:filename[.ext]]`

Converte il file TXT specificato in input, nel formato PRG. Se non viene specificata nessuna estensione, .PRG verra' utilizzata come default.

`-txt[:filename[.ext]]`
Fa esattamente l'opposto... ☺

Vediamo un esempio pratico: **`C64List CHAR.txt -prg:CHAR`**

C64List creera' un file di nome CHAR.PRG, che conterra' il codice BASIC presente nel file CHAR.TXT, in formato compatibile per il C64.Il file CHAR.PRG risultante potra' quindi essere caricato in CCS64 tramite il comando **File** -> **BASIC** -> **Load Program**.

DOVE REPERIRLO...

C64List, scritto da Jeff Hoag, puo' essere scaricato da Commodore Server: https://www.commodoreserver.com/Downloads.asp

CREDITS TO JEFF HOAG
c64List@gmail.com

Ridisegnamo i caratteri del C64

di Francesco Fiorentini

Scopriamo come rimappare i tasti del Commodore 64 con un semplice programma in BASIC.

***POKE** - scrive il valore indicato nella locazione di memoria specificata.*

***PEEK** - legge il contenuto della locazione di memoria indicata.*

***DATA** e **READ** verranno approfonditi successivamente...*

Come forse i piu' attenti avranno gia' intuito dall'editoriale, sono un ex possessore di Commodore 64 ed Amiga; quindi quale miglior modo di cominciare quest'avventura se non con un bell'articolo dedicato al biscottone?

Il Commodore 64, a differenza di altri computer, non possedeva nel proprio BASIC, Versione 2, delle primitive grafiche avanzate, costringendo quindi i programmatori e gli smanettoni ad ingegnarsi e studiarne a fondo l'architettura per riuscire anche nelle operazioni piu' semplici come disegnare un cerchio. A tal proposito, mi sono spesso chiesto come facessero alcune avventure testuali a riproporre a video dei caratteri diversi dai soliti; ecco una delle risposte.

Il Commodore 64 mappa la memoria video, quella cioe' che vediamo sul monitor, tra gli indirizzi esadecimali $0400-$07FF (1024-2047 in decimale) vedi tabella sotto:

$0400-$07FF	**1024-2047**	**Screen memory**
$0400-$07E7	1024-2023	Visible screen memory (1000 bytes)
$07E8-$07F7	2024-2039	Unused (16 bytes)
$07F8-$07FF	2040-2047	Area for sprite pointers (8 bytes)

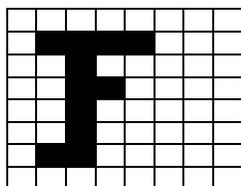
La memoria compresa tra gli indirizzi $0400-$07E7 (in decimale 1024-2023) definisce l'area visualizzata sullo schermo. Ricordo che il Commodore 64 ha una risoluzione nativa di 40 colonne per 25 righe. Moltiplicando 40x25 si ottiene giustamente il valore 1000 che e' l'esatta dimensione in bytes dell'area compresa tra gli indirizzi sopra citati.

Provate ad eseguire questo semplice listato:

```
10 for i=0 to 255
20 poke 1024+i, i
30 next i
```

Riuscite ad immaginare il risultato? Esatto, stampa a video tutti i 256 caratteri del C64, partendo dall'angolo a sinistra in alto. L'istruzione POKE infatti dice all'interprete BASIC di scrivere in memoria, all'indirizzo passato come primo parametro, il valore passato come secondo parametro. Semplice, vero?

I caratteri che abbiamo appena stampato sono memorizzati nella memoria ROM (Read only Memory) e non possono dunque essere modificati. Come facciamo quindi per cambiarli? Semplice, possiamo istruire il C64 a cercare questi valori da un'altra parte, per esempio in RAM! Facile? Non proprio, ma nemmeno troppo complicato.

Prima di addentrarci in questa operazione, vediamo pero' come ridisegnare un nuovo carattere. Ogni carattere e' definito da una griglia di 8 per 8. Ogni casella rappresenta un bit, valorizzata ad 1 se attivata, altrimenti a 0 (zero), mentre la riga intera e' un byte.

L'esempio sopra riportato equivale quindi a:

```
00000000 = (0)
01111000 = (120)
00100000 = (32)
00110000 = (48)
00100000 = (32)
00100000 = (32)
01100000 = (96)
00000000 = (0)
```

Prima pero' di sovrascrivere i caratteri, dobbiamo farne una copia, altrimenti l'interprete BASIC smettera' di funzionare. ☺ C'e' anche un altro piccolo problema da affrontare; la locazione ROM da cui vogliamo copiare e' inaccessibile per i programmatori, dobbiamo quindi renderla accessibile tramite l'istruzione:

```
poke 1, peek(1) and 251
```

Questa istruzione e' solo apparentemente semplice, infatti deve essere spiegata per essere compresa. Il contenuto della locazione di memoria 1 determina lo stato delle operazioni di I/O del C64 ed il bit 2 agisce come interruttore del modo in cui la CPU gestisce il set di caratteri ROM (character set ROM). Se il bit 2 e' settato a 0 (zero) la CPU permette ai dispositivi di I/O di raggiungere il character set ROM. Gli altri bit della locazione 1 controllano altre funzioni speciali del sistema e non vogliamo assolutamente alterarli!
Per agire solo sul bit 2 senza intaccare gli altri registri ricorreremo agli operandi AND e OR.

Assumiamo che il valore della locazione 1 sia:

bit	7	6	5	4	3	2	1	0
-	0	1	1	0	1	1	1	1

Non importa quale sia il valore di partenza, con un operazione di AND ed il valore 0 (zero) al bit 2 otterremo sempre il risultato di azzerare il bit 2 della locazione 1. Effettuando una operazione di AND con il valore 251, in binario 11111**0**11, otterremo:

valore	0	1	1	0	1	1	1	1
and 251	1	1	1	1	1	0	1	1
risultato	0	1	1	1	1	0	1	1

Allo stesso modo per riportare il valore ad 1, ricorreremo ad una operazione OR con il valore 4, in binario 00000**1**00:

valore	0	1	1	0	1	0	1	1
or 4	0	0	0	0	0	1	0	0
risultato	0	1	1	1	1	1	1	1

Il conseguente comando, che troveremo piu' avanti nel codice, sara' quindi:

```
poke 1, peek(1) or 4
```

L'operazione finalmente assume un senso ☺: scrivi nella locazione di memoria 1 il valore della locazione di memoria 1, combinato con la maschera AND 251. Affascinate vero? A voi il compito di decifrare la seconda…

A questo punto possiamo finalmente passare a copiare i caratteri:

```
fori=0to255*8
poke 14336+i, peek(53248+i)
next i
```

Si tratta di 256 caratteri (da 0 a 255) ed ogni carattere e' formato da 8 byte. Dobbiamo quindi copiare 256*8 byte (ciclo for da 0 a 255*8) richiedendo molto tempo.

La locazione di memoria originale in ROM e' all'indirizzo 53248, mentre la destinazione in RAM e all'indirizzo 14336.

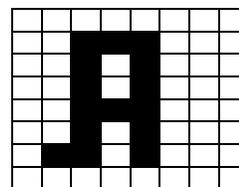
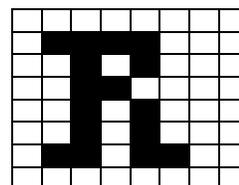
Ma, per agire sulla mappa di memoria senza interruzioni dobbiamo assicurarci che gli Interrupt del C64 non interferiscano con questa operazione, quindi provvederemo a disabilitarli tramite un'opportuna istruzione di mascheratura:

```
poke 56334, peek(56334) and 254
```

Questo fa in modo che le operazioni di I/O non interferiscano con la nostra copia. Una volta portata a ternine l'operazione per riabilitare nuovamente gli Interrupt:

```
poke 56334, peek(56334) or 1
```

Ormai avete la chiave di lettura di queste istruzioni; ma se per interesse volete approfondire cosa faccia la locazione di memoria 56334, vi rimando al link sotto: http://sta.c64.org/cbm64mem.html.

Ora che abbiamo capito come copiare i caratteri, passiamo a ridefinirli. Per puro ego mi limitero' a ridefinire le prime tre lettere del mio nome… ☺ La F l'abbiamo gia' disegnata, passiamo quindi alla R ed alla A.

Perdonatemi, ma come grafico faccio schifo, ne sono consapevole! ☹

Vi risparmio i binari; ecco i valori in byte:
F = 0, 120, 32, 48, 32, 32, 96, 0
R = 0, 120, 40, 48, 40, 40, 108, 0
A = 0, 56, 40, 40, 56, 40 ,104, 0

A questo punto non ci resta altro che andare a scrivere nelle locazioni di memoria che contengono i caratteri F, R, ed A i byte che abbiamo appena calcolato.

Per comodita' utilizzeremo i comandi DATA e READ. Questi comandi vanno spesso a braccetto con il comando RESTORE. Vediamo quindi cosa fanno in concreto prima di usarli!

Il comando **DATA** (vedi sintassi):

```
DATA <Dati> [, <Dati>...]
```

permette di immagazzinare in memoria una serie di dati statici; molto comodo per noi che dobbiamo passare una serie di valori consecutivi.

Il comando **READ** (vedi sintassi):

```
READ <Var> [, <Var>...]
```

si occupa invece di leggere i valori immagazzinati dal comando DATA e ad ogni singola lettura sposta il puntatore al dato successivo.

Il comando RESTORE invece inizializza il puntatore di READ alla prima occorrenza del comando DATA. Ecco spiegato l'arcano di quel singolo RESTORE nella prima linea di codice. ☺

I caratteri che vogliamo ridefinire, FRA sono rispettivamente alla posizione 6, 18 e 1 e verranno passati alla funzione di ridefinizione tramite la variabile C.

Facciamo l'esempio della lettera F:

```
data 0, 120, 32, 48, 32,
32, 96, 0

c=6
for i=0 to 7
read a
poke 14336+i+c*8, a
next i
```

Perche' il valore delle variabile C viene moltiplicato per 8? Come dicono gli anglosassoni, buona domanda.

Vi ricordate che ogni carattere e' composto da 8 byte, quindi come per la loro copia, quando andiamo a disegnarli dobbiamo riscrivere tutti e 8 i byte.

A questo punto abbiamo fatto tutto, non ci resta che provare a stampare a video i nostri nuovi caratteri.

```
Print "FRA   ARF   RAF"
```

Se tutto e' andato bene i dovreste vedere stampati a video i nostri caratteri ridefiniti. Bello vero?

Ecco il listato di pronto per essere eseguito sul C64:

```
10 RESTORE
20 PRINT CHR$(147)
25 REM DISABILITA GLI IRQ
30 POKE 56334, PEEK(56334) AND 254
40 POKE 1, PEEK(1) AND 251
50 PRINT "COPIA DEI CARATTERI"
60 FOR I = 0 TO 255*8
70 POKE 14336+i, PEEK(53248+i)
80 NEXT I
90 POKE 1, PEEK(1) OR 4
95 REM ABILITA GLI IRQ
100 POKE 56334, PEEK (56334) OR 1
105 REM USA LA MEMORIA DA $3800 DEC. 14336
110 POKE 53272, PEEK (53272) OR 14
120 PRINT "IL C64 ADESSO USA I CARATTERI $3800"
130 REM LETTERE RIDEFINITE F, R, A
140 DATA 0, 120, 32, 48, 32, 32, 96, 0
160 DATA 0, 120, 40, 48, 40, 40, 108, 0
170 DATA 0, 56, 40, 40, 56, 40 ,104, 0
190 REM F MAIUSCOLA
200 C=6
210 GOSUB 400
215 REM R MAIUSCOLA
220 C=18
230 GOSUB 400
235 REM A MAIUSCOLA
240 C=1
250 GOSUB 400
300 REM STAMPIAMO I CARATTERI RIDEFINITI
310 PRINT:PRINT:PRINT
320 PRINT "FRA   ARF   RAF"
390 END
400 REM CREAZIONE DEI NUOI CARATTERI
410 FOR I=0 TO 7
420 READ A
430 POKE 14336+I+C*8, A
440 NEXT I
450 RETURN
```

Ovviamente il codice riportato e' solo un esempio senza pretese, per mettere in pratica quanto spiegato l'articolo; non ha nessuna pretesa di essere utile in un gioco ne' tantomeno in un'applicazione. Come si suol dire, per puro scopo didattico.

A questo punto avete in mano la chiave di volta per ridefinire l'intero set di caratteri del Commodore 64 come piu' vi aggrada.

La palla adesso e' nelle vostre mani, il limite e' solo la vostra fantasia ...ed una griglia di 8*8. Per i piu' pigri,iIn rete si trovano decine di esempi di caratteri ben fatti, alcuni liberamente scaricabilii ed altri protetti da Copyright, ma da cui potete prendere spunto...

Se avetre domande, suggerimenti, o ritenete che abbia scritto qualcosa di errato, fatemelo sapere.

(Francesco)

INTERNATIONAL KARATE PLUS (IK+)

Colpi ben assestati

IK+ permette di assestare dei colpi niente male! Niente mosse speciali e fronzoli vari, i colpi sono reali e si percepiscono tali. Sembra quasi una simulazione piuttosto che un arcade.

Effetti speciali

Oltre ad una giocabilita' ed un concept invidiabili IK+ si contraddistingue anche per una serie di tocchi di classe apprezzabili.

GIUDIZIO SUL GIOCO

GIOCABILITA'

90%

Che dire, per me e' un capolavoro assoluto. Bello a vedersi, veloce, fluido e giocabile. Un MUST HAVE!

LONGEVITA'

80%

Nonostante siano passati quasi 30 anni, IK+ regge alla grande la prova del tempo. Rimane un capolavoro anche nel 2017. Io mi sono divertito tuttora a giocarlo!

La schermata del titolo ci accompagna con una musica che e' un tributo al Paula. Sicuramente e' tra le mie musiche preferite tra i giochi AMIGA.

International Karate Plus (IK+)

System 3 - Anno 1988 - Piattaforma Amiga

Signore e signori, tanto di cappello! International Karate Plus, IK+ per gli amici, e' un autentico capolavoro. E' il classico gioco che non ti delude mai. Anno dopo anno lo riprendo in mano per giocarci e mi stupisco di nuovo per come e' stato programmato. Ma andiamo con ordine e vediamo perche'.

Il concept del gioco e' piuttosto semplice. 3 Karateki si sfidano a duello a colpi di calci, pugni, sgambetti e testate. A secondo della precisione e della violenza del colpo assestato verremo ricompensati con 1 o 2 punti. Il primo che riesce a mettere a segno colpi tali da raggiungere 6 punti, vince l'incontro. Il secondo classificato, passa al livello successivo. Il terzo classificato invece, viene ripreso dal burbero maestro Kawasaki San (nome puramente inventato) e costretto ad una sessione addizionale di sfiancante allenamento.

Va da se che, nel caso il giocatore sia il terzo classificato, verra' estromesso dal torneo per ricominciare a lottare dalla cintura bianca. Eh, si', come nella migliore tradizione del Karate, il progressivo avanzamento del giocatore viene misurato con le cinture. Si parte dalla cintura bianca, per arrivare attraverso le gialla, verde, viola e marrone alla tanto agognata cintura nera; il simbolo indiscusso del karateka provetto.

Il gioco impressiona positivamente per la rapidita', la precisione delle collisioni dei colpi e la fluidita'. Un errato posizionamento infatti puo' determinare un colpo a vuoto e la successiva risposta dell'avversario.

IK+ e' il successore di International Karate, con il quale condivide parte del gameplay. Quello che pero' distingue i due giochi e' l'aggiunta in IK+ del terzo opponente. Puo' sembrare un dettaglio da poco, ma se ci fermiamo a riflettere non e' proprio cosi'. In molti giochi di combattimento dell'epoca possiamo piazzare qualche buon colpo e schivando gli attacchi dell'avversario portare in fodo il combattimento da vincenti. In IK+ no. Possiamo venire eliminati anche non avendo subito nessun colpo se gli avversari se le sono suonate di ragione! Questa meccanica di gioco impone una tattica decisamente improntata all'attacco!

Abbiamo appurato che la giocabilita' e' al top, ma per quanto riguarda grafica e sonoro? La grafica non e' niente di impressionante per l'Amiga, anche se e' decisamente ben fatta. Colorata al punto giusto, si lascia apprezzare, ma non fa certo balzare sulla sedia. Un discorso a parte lo merita invece il comparto sonoro. Per me indubbiamente una delle colonne sonore meglio riuscite su Amiga. Ammetto che a volte lo carico solo per ascoltarne la musica.

Ad essere pignoli un difetto in IK+ c'e': un solo sfondo! Avremmo preferito un po' piu' di varieta', anche se fortunatamente i programmatori si sono divertiti ad inserire un po' di effetti speciali e di animazioni per movimentare la scena. Avete per caso notato Pacman? Il ragnetto? Ed il sottomarino?

A questo punto non mi resta altro che augurarvi buon divertimento con IK+. (Francesco)

Sui prossimi numeri...

Eccoci giunti alla fine del primo numero. Mi auguro che la lettura, sia stata di vostro gradimento, almeno quanto lo e' stato per me la preparazione.

Devo ammettere che non e' uno scherzo lavorare da solo ad un progetto cosi' ambizioso, per questa ragione sono in cerca di collaboratori.

Avevo postato una richiesta di collaborazione su un gruppo Facebook, ma ho deciso comunque di provare a stilare il primo numero da solo per rendere l'idea di quello che vorrei realizzare con questo progetto.

Sono della convinzione che un'azione valga molto piu' di tante parole e quindi ecco a voi pronto il primo numero di RetroMagazine!

Il futuro di questa avventura e' anche nelle vostre mani, piu' collaborazione ottengo, piu' articoli verranno aggiunti ad ogni numero e di conseguenza anche il numero delle pagine sara' destinato ad aumentare.

La mia esperienza si basa principalmente su macchine Commodore, quindi non stupitevi se molti articoli saranno incentrati su C64 e Amiga. Cio' non toglie che provero' anche a scrivere di altri sistemi. In questo caso pero' ogni aiuto sara' piu' che benvenuto.

Ma vediamo adesso cosa ho in mente per il futuro di RetroMagazine:

Quanti voi hanno sentito parlare di **GEOS**? Esatto proprio lui, il Graphic Environment Operating System che tanto ha stupito chi ha avuto la fortuna di provarlo sul proprio C64 nel lontano 1986. Mi piacerebbe dedicargli un articolo. Che ne dite?

Proprio in questi giorni **Davide Fichera**, utente del gruppo **Commodore 64/128 Italia** ha rilasciato un gioco interamente sviluppato in BASIC per il Commodore 64. Ebbene mi sono messo in contatto con lui che ha gentilmente accettato di scrivere un articolo in cui oltre a proporre il listato del gioco, con relativi commenti, ci spiegara' la motivazione che lo ha spinto a scrivere il gioco. ☺

Mi piacerebbe dedicare un po' di spazio anche agli emulatori. Alcuni sono immediati, ma altri richiedono conoscenze specifiche per essere maneggiati correttamente, vero **WinUAE**?

Per adesso mi fermo, non voglio svelare troppo... anche perche' non so quando il numero due sara' pronto per essere distribuito. ☺

Se volete scrivermi per suggerimenti ed idee:
fiorentini.francesco.it@gmail.com

RetroMagazine
Anno 1 - Numero 1

Direttore Responsabile
Fiorentini Francesco

Ottobre 2017